REMARKS

In response to the Final Office Action mailed September 24, 2007, Applicants respectfully request reconsideration. Claims 1-8 and 12 were previously pending in this application. By this amendment, no claims has been amended. As a result, claims 1-8 and 12 are pending for examination with claims 1, 5, 7, and 8 being independent. No new matter has been added.

Rejections Under 35 U.S.C. §103

The Office Action rejected claims 1-8 and 12 under 35 U.S.C. 103(a) as allegedly being unpatentable over Hadjiyiannis et al., "ISDL: An Instruction Set Description Language for Retargetabillity" (hereinafter Hadjiyiannis), further in view of Vos, GB 2,127,188 (hereinafter Vos) and further in view of A. N. Edmonds "Microcoding and bit-slice techniques" (hereinafter Edmonds). Applicant respectfully disagrees.

Claim 1 recites:

> An assembler for a target microprocessor to automatically track changes in an instruction set of the target microprocessor, the assembler comprising:
>
> a descriptor file containing information descriptive of the instruction set of said target microprocessor, wherein information about a starting position of at least one bit field and a number of bits available for the at least one bit field for at least one instruction in the instruction set is derived and stored as constraints due to the instruction set;
>
> a translation device of the assembler for translating assembly language instructions into machine language as an output;
>
> a fetching device of the assembler for deriving information from the assembly language instructions and acquiring data from said descriptor file using the information, wherein the data is representative of the constraints due to the instruction set and includes selected data comprising at least one encoding function for the at least one instruction due to the changes in the instruction set;
>
> a control device of the assembler arranged to receive said data from said fetching device and said machine language from said translation device, and operable to constrain the machine language to conform to an architecture of said instruction set using the data received from the fetching device; and
>
> a data transfer device of the assembler arranged to output the selected data fetched from said descriptor file directly to a linker to enable the linker to perform operations on the constrained machine language using the selected data, including operations on external symbols using the at least one encoding function.

On page 6, the Office Action alleges that Hadjiyiannis discloses "an assembler for a target microprocessor to automatically track changes in an instruction set of the target microprocessor, the assembler comprising ..." "tracking the changes is interpreted as being representative of constrains due to the instruction set architecture provided by the Instruction Set Description Language" of Hadjiyiannis. Support for the limitation "an assembler for a target microprocessor to automatically track changes in an instruction set of the target microprocessor" can be found on page 8, lines 31-32, of the present specification. Hadjiyiannis does not disclose **automatically** tracking changes in an instruction set of the target microprocessor. (Emphasis added).

Further, Hadjiyiannis does not disclose **an assembler** for a target microprocessor to **automatically track changes** in an instruction set of the target microprocessor. (Emphasis added). Hadjiyiannis is directed to a language called ISDL (Instruction Set Description Language) that can be used to describe target architectures in a re-targetable compiler. The ISDL description of a target architecture can be used to generate an assembler (Hadjiyiannis, Section III, paragraph 1, lines 7-8). Hadjiyiannis discusses an automatic **assembler generator** that receives an ISDL description as input, and **produces an assembler** which assembles the compiler's output to a binary file (Hadjiyiannis, Section V, paragraph 2, lines 1-3). (Emphasis added). The assembler of Hadjiyiannis does not **automatically track changes** in an instruction set of the target microprocessor. (Emphasis added). Rather, the assembler of Hadjiyiannis appears to be fixed. If any changes occur, the ISDL description of a target architecture can be used to generate **another** assembler, as opposed to providing "an assembler for a target microprocessor to automatically track changes," as recited in claim 1.

On pages 6-7, the Office Action alleges that Hadjiyiannis discloses "assembler comprising: a descriptor file containing information descriptive of the instruction set of said target microprocessor, wherein information about a starting position of at least one bit field and a number of bits available for the at least one bit field for at least one instruction in the instruction set is derived and stored as constraints due to the instruction set." As discussed above, in Hadjiyiannis, **the ISDL description of a target architecture can be used to generate an assembler**. (Emphasis added). Therefore, description of a target architecture is used to generate assembler of Hadjiyiannis. In contrast, *an assembler* recited in claim 1 *comprises* a descriptor file. (Emphasis added).

On page 8, the Office Action states that "the compiler is part of the claimed assembler." Applicant respectfully disagrees.

On page 9, the Office Action concedes that Hadjiyiannis does not disclose "a data transfer device of the assembler arranged to output the selected data fetched from said descriptor file directly to a linker to enable the linker to perform operations on the constrained machine language using the selected data, including operations on external symbols using the at least one encoding function," as recited in claim 1. The Office Action goes on to allege that Vos discloses this limitation. However, Vos does not discuss fetching "from said descriptor file" wherein a descriptor file is a file "containing information descriptive of the instruction set of said target microprocessor," as recited in claim 1. Vos discusses the ICS integration source file (block 6) provides a concise, **human-readable description** of the hardware/software interface. (Vos, page 2, lines 38-40). (Emphasis added).

On page 10, the Office Action concedes that nether Hadjiyiannis nor Vos discloses "... a starting position ..." or "using the information, wherein the data is representative of the constraints due to the instruction set and includes selected data comprising at least one encoding function for the at least one instruction due to the changes in the instruction set." The Office Action then alleges that Edmonds discloses these features. Edmonds does discuss that to be able to produce code for any destination hardware, meta assemblers first need to be fed with specific information about the hardware configuration. (Edmonds, page 266). Edmonds describes the pipelining skews in the definition file. However, Edmonds does discuss that the data is representative of the constraints due to the instruction set and includes selected data comprising at least one encoding function for the at least one instruction due to the changes in the instruction set.

In view of the forgoing, claim 1 patentably distinguishes over Hadjiyiannis, Vos, and Edmonds, either alone or in combination.

Claims 2-4 and 12 depend from claim 1 and are allowable for at least the same reasons.

Accordingly, withdrawal of the rejection of claims 1-4 and 12 is respectfully requested.

Claim 5 recites:

A method of assembling a machine language program for a target microprocessor comprising:

providing a descriptor file containing information descriptive of an instruction set of said target microprocessor, wherein information about a starting position of at least one bit field and a number of bits available for the at least one bit field for at least one instruction in the instruction set is derived and stored as constraints due to the instruction set;

translating assembly language instructions into machine language wherein the translation comprises:

directly transliterating the assembly language instructions to machine language;

deriving information from the assembly language instructions and acquiring data from said descriptor file using the information, wherein the data is representative of the constraints due to the instruction set and includes selected data comprising at least one encoding function for the at least one instruction due to changes in the instruction set;

constraining the directly transliterated machine language to conform to an architecture of said instruction set, thereby assembling the machine language program for the target microprocessor, using the data acquired from the descriptor file; and

transferring the selected data acquired from said descriptor file directly to a linker to enable the linker to perform operations on the constrained machine language using the selected data, including operations on external symbols using the at least one encoding function.

It should be appreciated from the above discussion that Hadjiyiannis, Vos, and Edmonds do not disclose or suggest "providing a descriptor file containing information descriptive of an instruction set of said target microprocessor, wherein information about a starting position of at least one bit field and a number of bits available for the at least one bit field for at least one instruction in the instruction set is derived and stored as constraints due to the instruction set; translating assembly language instructions into machine language wherein the translation comprises: directly transliterating the assembly language instructions to machine language; deriving information from the assembly language instructions and acquiring data from said descriptor file using the information, wherein the data is representative of the constraints due to the instruction set and includes selected data comprising at least one encoding function for the at least one instruction due to changes in the instruction set; constraining the directly transliterated machine language to conform to an architecture of said instruction set, thereby assembling the machine language program for the target microprocessor, using the data acquired from the descriptor file; and transferring the selected data acquired from said descriptor file directly to a

linker to enable the linker to perform operations on the constrained machine language using the selected data, including operations on external symbols using the at least one encoding function," as recited in claim 5.

In view of the foregoing, claim 5 patentably distinguishes over Hadjiyiannis, Vos, and Edmonds, either alone or in combination.

Claim 6 depends from claim 5 and is allowable for at least the same reasons.

Accordingly, withdrawal of the rejection of claims 5 and 6 is respectfully requested.

Claim 7 recites:

> A method of preparing a program executable on a target microprocessor comprising:
>
> capturing data from an instruction set of said target microprocessor thereby forming a descriptor file containing information descriptive of said instruction set, wherein information about a starting position of at least one bit field and a number of bits available for the at least one bit field for at least one instruction in the instruction set is derived and stored as constraints due to the instruction set;
>
> providing assembly language instructions for said target microprocessor;
>
> translating each assembly language instruction into a corresponding machine language output;
>
> deriving information from the assembly language instructions and acquiring data from the descriptor file using the information, wherein the data is representative of the constraints due to the instruction set and includes selected data comprising at least one encoding function for the at least one instruction due to changes in the instruction set;
>
> using the data acquired from said descriptor file, constraining the machine language output to conform to an architecture of said instruction set; and
>
> transferring the selected data acquired from said descriptor file directly to a linker to enable the linker to perform operations on the constrained machine language using the selected data, including operations on external symbols using the at least one encoding function.

It should be appreciated from the above discussion that Hadjiyiannis, Vos, and Edmonds do not disclose or suggest "a method of preparing a program executable on a target microprocessor comprising: capturing data from an instruction set of said target microprocessor thereby forming a descriptor file containing information descriptive of said instruction set, wherein information about a starting position of at least one bit field and a number of bits available for the at least one bit field for at least one instruction in the instruction set is derived

and stored as constraints due to the instruction set; providing assembly language instructions for said target microprocessor; translating each assembly language instruction into a corresponding machine language output; deriving information from the assembly language instructions and acquiring data from the descriptor file using the information, wherein the data is representative of the constraints due to the instruction set and includes selected data comprising at least one encoding function for the at least one instruction due to changes in the instruction set; using the data acquired from said descriptor file, constraining the machine language output to conform to an architecture of said instruction set; and transferring the selected data acquired from said descriptor file directly to a linker to enable the linker to perform operations on the constrained machine language using the selected data, including operations on external symbols using the at least one encoding function," as recited in claim 7.

In view of the foregoing, claim 7 patentably distinguishes over Hadjiyiannis, Vos, and Edmonds, either alone or in combination.

Accordingly, withdrawal of the rejection of claim 7 is respectfully requested.

Claim 8 recites:

A method of preparing a program executable on a microprocessor, comprising:

providing plural program modules, at least one of said modules having one or more instructions including external symbols, wherein external symbols have values which cannot be determined without reference to another program module;

providing a descriptor file containing information descriptive of an instruction set of said target microprocessor, wherein information about a starting position of at least one bit field and a number of bits available for the at least one bit field for at least one instruction in the instruction set is derived and stored as constraints due to the instruction set;

translating assembly language instructions into machine language wherein the translation comprises:

directly transliterating the assembly language instructions into machine language;

deriving information from the assembly language instructions and acquiring data from said descriptor file using the information, wherein the data is representative of the constraints due to the instruction set and includes selected data comprising at least one encoding function for the at least one instruction due to changes in the instruction set; and

constraining the directly transliterated machine language to conform to an architecture of said instruction set using the data acquired from the descriptor file;

transferring the selected data acquired from said descriptor file directly to a linker to enable the linker to perform operations on the constrained machine language using the selected data; and

binding external symbols to addresses using the at least one encoding function from the selected data acquired from said descriptor file, thereby preparing the program executable on the microprocessor.

It should be appreciated from the above discussion that Hadjiyiannis, Vos, and Edmonds do not disclose or suggest "a method of preparing a program executable on a microprocessor, comprising: providing plural program modules, at least one of said modules having one or more instructions including external symbols, wherein external symbols have values which cannot be determined without reference to another program module; providing a descriptor file containing information descriptive of an instruction set of said target microprocessor, wherein information about a starting position of at least one bit field and a number of bits available for the at least one bit field for at least one instruction in the instruction set is derived and stored as constraints due to the instruction set; translating assembly language instructions into machine language wherein the translation comprises: directly transliterating the assembly language instructions into machine language; deriving information from the assembly language instructions and acquiring data from said descriptor file using the information, wherein the data is representative of the constraints due to the instruction set and includes selected data comprising at least one encoding function for the at least one instruction due to changes in the instruction set; and constraining the directly transliterated machine language to conform to an architecture of said instruction set using the data acquired from the descriptor file; transferring the selected data acquired from said descriptor file directly to a linker to enable the linker to perform operations on the constrained machine language using the selected data; and binding external symbols to addresses using the at least one encoding function from the selected data acquired from said descriptor file, thereby preparing the program executable on the microprocessor," as recited in claim 8.

In view of the foregoing, claim 8 patentably distinguishes over Hadjiyiannis, Vos, and Edmonds, either alone or in combination.

Accordingly, withdrawal of the rejection of claim 8 is respectfully requested.

CONCLUSION

A Notice of Allowance is respectfully requested. The Examiner is requested to call the undersigned at the telephone number listed below if this communication does not place the case in condition for allowance.

If this response is not considered timely filed and if a request for an extension of time is otherwise absent, Applicant hereby requests any necessary extension of time. If there is a fee occasioned by this response, including an extension fee, that is not covered by an enclosed check, please charge any deficiency to Deposit Account No. 23/2825.

Dated: December 21, 2007

Respectfully submitted,

By: /James H. Morris/

James H. Morris, Reg. No. 34,681
Wolf, Greenfield & Sacks, P.C.
600 Atlantic Avenue
Boston, Massachusetts 02210-2206
Telephone: (617) 646-8000